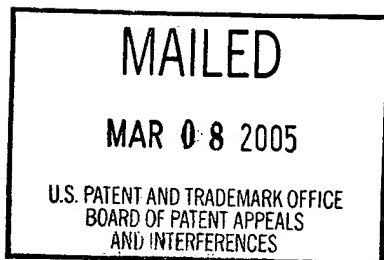The opinion in support of the decision being
entered today was **not** written for publication
and is **not** binding precedent of the Board.

Paper No. 24

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte HIROAKI KIMURA and MASAHIRO NOGUCHI

Appeal No. 2004-0558
Application No. 09/241,735

HEARD: February 22, 2005

Before: KRASS, GROSS and NAPPI, **Administrative Patent Judges**.

NAPPI, **Administrative Patent Judge**.

DECISION ON APPEAL

This is a decision on appeal under 35 U.S.C. § 134 of the final rejection of

claims 1 through 37, which constitute all the claims in the application.

Invention

The invention relates to a software analysis apparatus that generates

software analysis information. The software analysis information is sequentially

stored in an arbitrary unit or at an arbitrary timing, and when executing a

program analysis the information is readout from the storage. (See page 5 of

appellants' specification)

Claims 1 and 11 are representative of the invention and reproduced below:

1.   A software analysis apparatus comprising:
program analysis information generation means for automatically
generating program analysis information required for analyzing a
computer program;
program analysis information storage means for classifying the
program analysis information generated by said program analysis
information generation means in an arbitrary unit or at an arbitrary timing,
and sequentially storing the program analysis information in a
predetermined data recording medium; and
program analysis means for executing program analysis by reading
out the program analysis information from said data recording medium.

11.   A software analysis apparatus for generating program analysis
information required for analyzing a computer program, comprising:
means for hierarchically registering the generated program
analysis information in a database in units of analysis objectives;
means for implementing analysis of the hierarchically registered
program analysis information by reading out the program analysis
information already registered in a predetermined layer in
correspondence with an analysis objective upon analyzing the computer
program.

Reference

The reference relied upon by the examiner are:

Wygodny et al. (Wygodny)   6,282,701   August 28, 2001

Rejection at Issue

Claims 1 through 37 stand rejected under 35 U.S.C. § 112, first paragraph and claims 1 through 37 stand rejected under 35 U.S.C. § 102(e) as being anticipated by Wygodny. Throughout the opinion we make reference to the briefs[1] and the answer for the respective details thereof.

Opinion

We have carefully considered the subject matter on appeal, the rejections advanced by the examiner and the evidence of lack of enablement and anticipation relied upon by the examiner as support for the rejections. We have, likewise, reviewed and taken into consideration, in reaching our decision, the appellants' arguments set forth in the briefs along with the examiner's rationale in support of the rejections and arguments in rebuttal set forth in the examiner's answer.

With full consideration being given to the subject matter on appeal, the examiner's rejections and the arguments of appellants and the examiner, and for the reasons stated *infra,* we will not sustain the examiner's rejection of claims 1 through 37 under 35 U.S.C. § 112,first paragraph, nor will we sustain the examiner's rejection of claims 1 through 37 under 35 U.S.C. § 102.

---

[1] Appellants filed an Appeal Brief on March 12, 2003 and a Reply Brief on July 28, 2003.

Rejection of Claims under 35 U.S.C. § 112, first paragraph

Appellants argue, on pages 9 through 11 of the brief, that the examiner's objection to the specification for incorporating by reference "essential material" is unreasonable. On page 5 of the reply brief, appellants argue that the issue is intertwined with the rejection based upon 35 U.S.C. § 112, first paragraph, that the objection is within the jurisdiction of the Board of Patent Appeals and Interferences.

We disagree. We find this is a petitionable matter and as such we will not consider the arguments presented on pages 9 through 11 of the brief and page 5 of the reply brief directed to the examiner's objection to the specification.

On pages 9 and 12 of the brief appellants state that they are unsure if the rejection is based upon the written description or enablement provision of 35 U.S.C. § 112, first paragraph. On pages 15 through 17 of the brief appellants argue the rejection as if it were a written description and on pages 17 through 20 of the brief appellants argue the rejection as if it were an enablement rejection.

In the answer, on page 6, the examiner clarifies that the rejection is based on the enablement provision of 35 U.S.C. § 112, first paragraph. Further, on page 9 of the answer, the examiner states:

> The examiner has concurred that certain processes relating to program analysis of elements such as source code, symbol table, call graph, flow graph, and data flow information are obvious and would have been known to one skilled in the art. However, examiner has asserted that processes such as **metrics information**, **redundancy information** and

maintenance document information do not have an equivalent specific meaning in the art and are not sufficiently defined by the specification. (Emphasis original.)

The examiner also acknowledges that the appellants' specification includes a definition of these terms. However the examiner asserts that these definitions are insufficient.

We disagree with the examiner. Initially we note that none of independent claims 1, 11, 13, 22, 24 or 33 specifically claims the program analysis information asserted by the examiner as not being enabled. Each of the independent claims broadly claims "program analysis information generation" which we find to be broad enough to encompass source code, symbol table, call graph, flow graph, and data flow information, which both appellants and the examiner have agreed is well known. Thus, by the examiner's own statements, the rejection of claims 1, 11, 13, 22, 24 and 33 is improper.

We find that the limitations directed to metrics information, redundancy information and maintenance document information, the program analysis information which the examiner is asserting as not being enabled, are only present in claims 4, 16, and 27. Appellants argue, on page 37 of the brief (Appendix B), that metrics information is defined on page 13 of the originally filed specification, and redundancy information and maintenance document information are defined on page 14 of the originally filed specification. The test

for enablement under the first paragraph of 35 U.S.C. § 112, is whether one reasonably skilled in the art could make or use the claimed invention from the disclosed subject matter together with information in the art, without undue experimentation. United States v. Telectronics, Inc., 857 F.2d 778, 785, 8 USPQ2d 1217, 1223 (Fed. Cir. 1988), cert denied, 490 U.S. 1046 (1989). A disclosure can be enabling even though some experimentation is necessary. Hybritech Inc. v. Monoclonal Antibodies, Inc., 802 F.2d 1367, 1384, 231 USPQ 81, 94 (Fed. Cir. 1986), cert denied, 480 U.S. 947 (1987). We find that the descriptions, on pages 13 and 14 of the originally filed specification, of these forms of program analysis information provide adequate description that one of the art could produce the claimed device without undue experimentation.

For the forgoing reasons we will not sustain the examiner's rejection of claims 1 through 37 under 35 U.S.C. § 112, first paragraph.

Rejection of Claims 35 U.S.C. § 102

Appellants argue, on page 18 of the brief, that the rejection based upon Wygodny is improper as:

> *[C]ontrary to the Examiner's assertion, "trace" as used in Wygodny et al., and "analysis information" in the present invention are technically different concepts.* (Emphasis original.)
> The disclosed invention uses program analysis information obtained from analysis of a non-executing software program.

On page 19 of the brief, appellants argue "[t]he program analysis information of

the disclosed and claimed invention, such as syntactic analysis tree, call graph,

and flow graph, is statically obtained from source code, *but is not generated*

during execution of the program." (Emphasis original.)

> In reply, the examiner states, on page 12 of the answer,
>
> A "trace", as disclosed by Wygodny, and as known in the art, merely
> represents a sample of **analysis information** gathered from a program
> (software) that has been stored in computer memory. The examiner
> asserts that the claims do not recite limitations that distinguish the
> claimed inventions analysis information from the trace information of
> Wygodny. Appellants have argued that prior art does not teach program
> analysis information being obtained from a non-executing software
> program. The examiner asserts that there are *no claimed limitations*
> which would indicate that the obtained (analyized [sic]) program
> information is gathered from a "non-executing software program."
> (Emphasis original.)

We concur with the examiner's claim interpretation. Claims will be given their

broadest reasonable interpretation consistent with the specification, but

limitations appearing in the specification will not be read into the claims. *In re*

Etter 756 F.2d 852, 858, 225 USPQ 1, 5 (Fed. Cir. 1985). In analyzing the

scope of the claim, office personnel must rely on the appellant's disclosure to

properly determine the meaning of the terms used in the claims. *Markman v.*

Westview Instruments, Inc., 52 F3d 967, 980, 34 USPQ2d 1321, 1330 (Fed.

Cir. 1995). "[I]nterpreting what is *meant* by a word in a claim 'is not to be

confused with adding an extraneous limitation appearing in the specification,

which is improper.'" (emphasis original) *In re Cruciferous Sprout Litigation,* 301 F.3d 1343, 1348, 64 USPQ2d 1202, 1205, (Fed. Cir. 2002) (citing *Intervet America Inc v. Kee-Vet Laboratories Inc.* 12 USPQ2d 1474, 1476 (Fed. Cir. 1989)). "[T]he terms used in the claims bear a "heavy presumption" that they mean what they say and have the ordinary meaning that would be attributed to those words by persons skilled in the relevant art." *Texas Digital Sys, Inc. v. Telegenix, Inc.,* 308 F.3d 1193, 1202, 64 USPQ2d 1812, 1817 (Fed. Cir. 2002). "Moreover, the intrinsic record also must be examined in every case to determine whether the presumption of ordinary and customary meaning is rebutted." (citation omitted). "Indeed, the intrinsic record may show that the specification uses the words in a manner clearly inconsistent with the ordinary meaning reflected, for example, in a dictionary definition. In such a case, the inconsistent dictionary definition must be rejected." *Texas Digital Systems, Inc. v. Telegenix, Inc.,* 308 F.3d 1193, 1204, 64 USPQ2d 1812, 1819 (Fed. Cir. 2002). Absent claim language carrying a narrow meaning, the PTO should only limit the claim based upon the specification or prosecution history when those sources expressly disclaim the broader definition. *In Re Bigio,* 381 F3d 1320, 1325, 72 USPQ2d 1209, 1211 (Fed. Cir. 2004) (citing *Liebel-Farcheim Co, v. Medrad, Inc.,* 358 F3d 898, 906-09 (Fed. Cir. 2004).

Claim 1 includes the limitation "program analysis information generation means for automatically generating program analysis information required for analyzing a computer program." Appellants argue on page 19 of the appeal brief that program analysis information is obtained from a non-executed software program. We do not consider this to be an express disclaimer of the broader interpretation of program analysis information asserted by the examiner (that program analysis information includes information from executed program code) in view of the narrower interpretation assertion by appellants, that program analysis information is "obtained from non-executed program code." Similarly, we do not find that the specification contains an express disclaimer of the broader interpretation. Thus, we find that the scope of the claim 1 limitation of "program analysis information generation means for automatically generating program analysis information required for analyzing a computer program" is not limited to only information obtained from non-executed program code.

With respect to claims 1 through 6, 8 through 10 and 35, appellants argue on page 20 of the brief:

> Wygodny et al. not only does not disclose generation of program analysis information, as that term would be understood by a reading of Appellants' disclosure, but Wygodny et al. also does not disclose program analysis information storage means for storing and classifying the program analysis information.

The examiner responds, on page 13 of the answer, stating:

> [A]ppellants have not disclosed specifically how the analysis data is classified. Wygodny discloses classifying data as a graphical representation of the collected (analyzed) data showing program flow, program calls, and I/O information. (Figs. 3A, 3B, 5-7, 13-14) Further, and as admitted by appellants (page 23, line 11), Figure 7 of Wygodny discloses a class page window that provides a hierarchal tree of trace objects sorted by class (i.e. **classification** of the analysis information). (Emphasis original.)

We disagree with the examiner's rationale. Claim 1 includes the limitation of "program analysis information storage means for classifying the program analysis information generated by said program analysis information generation means in an arbitrary unit or at an arbitrary timing, and sequentially storing the program analysis information in a predetermined data recording medium." Thus, we find that the claimed storage means performs two functions, classifying information in either an arbitrary unit or at an arbitrary timing and storing the information. As stated *supra*, we find that the trace information of Wygodny meets the claimed "program analysis information." Further, we find that Wygodny teaches that the trace information can be provided in real time (online mode, discussed in columns 6 and 7) as the program is executing or can be saved to a file for subsequent analysis (remote mode described in columns 5 and 6, column 6, lines 9 and 10 identify that the trace data is written to a file). However, we do not find that the data is classified in either an arbitrary unit or at an arbitrary timing. The classification, which the examiner is referring to in figures 3A, 3B, 5-7, 13-14, is the result of a trace filter. (See column 12, lines 32-43, column 13, lines 58-65, and column 14, lines 35-47). However, this

filtering is performed on the data displayed to the user in response to user input after the trace data is generated (the trace file is not modified). (See column 21, lines 26-31). Thus, we find that the classification performed by Wygodny's device is performed at a definite time, after the trace is performed and in units designated by the user, and not "in an arbitrary unit or at an arbitrary timing" as claimed by claim 1. As such, we do not find that Wygodny teaches that the claimed program analysis information storage means classifies and stores the program analysis information. Accordingly, we will not sustain the examiner's rejection of independent claim 1, or the claims dependent on claim 1, namely, claims 2 through 10 and 35.

On pages 20 and 21 of the brief, appellants argue that the rejection of independent claims 13 and 24 is improper as these claims also include the limitation of classifying and storing the program analysis information. We find that independent claims 13 and 24 contain limitations directed to classifying and storing the program analysis information similar to those discussed *supra* with respect to claim 1. Similarly, we will not sustain the rejection of independent claims 13 and 24, nor the claims dependent on claims 13 and 24, claims 14 through 21, 25 through 32, 36 and 37, for the reasons given *supra* with respect to claim 1.

Appellants argue, on pages 22 through 26 of the brief, that the rejection of

independent claims 11, 22 and 33 is improper, as Wygodny does not disclose

that the program analysis information is stored in a database. Specifically,

appellants argue, on page 22 of the brief, that:

> Wygodny et al. does not disclose a software analysis apparatus for
> generating program analysis information required for analyzing a
> computer program which includes among other features, "... means for
> hierarchically registering the generated program analysis information in a
> database in units of analysis objectives; [and] means for implementing
> analysis of the hierarchically registered program analysis information by
> reading out the program analysis information already registered in a
> predetermined layer in correspondence with an analysis objective upon
> analyzing the computer program", as recited in independent claim 11.

Appellants make similar arguments, on page 24 of the brief, with respect to

claims 22 and 33.

> The examiner's response, on page 14 of the answer, states:

> Appellants have also argued that prior art does not disclose a database or
> database structure. The examiner asserts that while this is obviously an
> inherent feature to any software program analysis system, it is also
> expressly disclosed by Wygodny in Fig. 1 (121). (Emphasis original.)

We are unconvinced by the examiner's rationale. We find that claim 11

includes the limitation of "hierarchically registering the generated program

analysis information in a database in units of analysis objects." Thus, claim 11

includes that the data is hierarchically registered in a database. We find that

independent claims 22 and 33 contain similar limitations. While we disagree

with the examiner that figure 1, item 121, is sufficient evidence to support the

conclusion that Wygodny discloses a database, we do find that Wygodny

teaches that the trace information is stored within a database. (See Column 8,

lines 11-15). However, we do not find that Wygodny teaches that the trace data

(program analysis information) is hierarchically registered in the database.

Accordingly, we will not sustain the examiner's rejection of independent claims

11, 22 and 33, nor the claims dependent on claims 11, 22 and 33, namely,

claims 12, 23, and 34.

In summary we will not sustain the examiner's rejections under either 35 U.S.C. § 112, or 35 U.S.C. § 102. Accordingly, we reverse the examiner's rejection of claims 1 through 37.

REVERSED

ERROL A. KRASS   )
Administrative Patent Judge   )
)
)
)
) BOARD OF PATENT
ANITA PELLMAN GROSS   )   APPEALS AND
Administrative Patent Judge   )   INTERFERENCES
)
)
)
)
ROBERT E. NAPPI   )
Administrative Patent Judge   )

REN/vsh

CONNOLLY, BOVE, LODGE & HUTZ LLP
SUITE 800
1990 M STREET NW
WASHINGTON DC 20036-3425